

CLAIMS

We claim:

1. A method of generating a dialog manager for a spoken dialog service, the method comprising:

selecting a top level flow controller;

selecting available reusable subdialogs below the top level flow controller, the reusable subdialogs being isolated from application dependencies;

developing a subdialog for each application part not having an available subdialog; and

testing and deploying the spoken dialog service using the selected top level flow controller, selected reusable subdialogs and developed subdialogs, wherein the top level flow controller, reusable subdialogs and developed subdialogs interact independent of their decision model.

2. The method of claim 1, wherein the available reusable subdialogs are selected from a group comprising: telephone number, social security number, account number, address, e-mail address and name.

3. The method of claim 2, wherein the reusable subdialogs manage mixed-initiative conversations with a user.

4. The method of claim 1, wherein an available reusable subdialog is an input subdialog.

5. The method of claim 4, wherein the available reusable subdialog further comprises a confirmation component.

6. The method of claim 4, wherein the reusable input subdialog handles silence, rejection, low confidence natural language understanding (NLU) scores and explicit information in an input dialog with a user.
7. The method of claim 1, wherein the top level flow controller is a recursive transition network (RTN) flow controller.
8. The method of claim 7, wherein the available reusable subdialogs are RTN flow controllers.
9. The method of claim 8, wherein the available reusable subdialogs are rule-based flow controllers.
10. The method of claim 8, wherein at least one state in the RTN flow controller has a subdialog attribute that is the name of a flow controller invoked as a subdialog.
11. The method of claim 10, wherein the state in the RTN flow controller having the subdialog attribute that invokes a subdialog further comprises a set of instructions that retrieve values from a parent dialog and set values in the invoked subdialog.
12. The method of claim 1, further comprising implementing a local context within a dialog data file associate with the dialog manager.
13. A method of generating a dialog manager for use in a spoken dialog service, dialog manager supporting context shifts in a spoken dialog, the method comprising:
 - selecting a top level dialog flow controller;
 - incorporating a context shift component;
 - selecting available reusable subdialogs for being invoked by the top level flow controller, the reusable subdialogs being isolated from application dependencies; and

testing and deploying the spoken dialog service using the selected top level flow controller and selected reusable subdialogs, wherein when a user of the system changes the context of the spoken dialog while in a reusable subdialog, the context shift component causes a parent dialog of the subdialog to be set to a state described by the context shift.

14. The method of claim 13, wherein during a spoken dialog, when a subdialog is invoked by a parent dialog, the context shift component causes the subdialog to inherit the context shifts of the parent dialog.

15. The method of claim 14, wherein when a user of the system changes the context of the spoken dialog while in a reusable subdialog, the context shift component further returns a message to the parent dialog that a context shift has occurred.

16. The method of claim 13, wherein a context shift is triggered by user input and generates a state name where the shift goes.

17. The method of claim 13, wherein the application dependencies are part of the top level flow controller.

18. The method of claim 17, wherein the top level flow controller and subdialogs interact independent of their decision models.

19. The method of claim 13, wherein the generated dialog manager supports chronological shifts in the dialog.

20. The method of claim 13, wherein the generated dialog manager supports digression in the dialog.